该文件旨在使 BigInteger 类在 C++ 中能够像基本数据类型一样使用。由于该类是在 vector< int > 类的基础上做计算，所以其计算速度不如 int，但也因此获得了更高的精度。

已重载的运算符

运算符类型	运算符
双目运算符	+(加), -(减), *(乘), /(整除), %(取模)
关系运算符	==(等于), !=(不等于), <(小于), >(大于), <=(小于等于), >=(大于等于)
逻辑运算符	||(逻辑或), &&(逻辑与), !(逻辑非)
单目运算符	+(正), -(负)
自增自减运算符	++(自增), –(自减)
赋值运算符	=, +=, -=, *=, /=, %=
位运算符	>>(右移运算符，与输入流关联), <<(左移运算符，与输出流关联)

支持的其他函数

函数声明	函数功能
size_t size() const	返回 BigInteger 对象的位数
BigInteger e(size_t n) const	返回 BigInteger 对象 $\times 10^n$ 后的值
BigInteger abs() const	返回 BigInteger 对象的绝对值

更多函数请使用者在已有代码基础上自行编写。

计算耗时比($\frac{t_{BigInteger}}{t_{int/longlong}}$)

运算符	int	long long
ostream& operator<<(ostream&, const T&)	1.087	1.097
istream& operator>>(istream&, T&)	1.355	1.211
abs()	1.407	1.273
比较运算符	1.490	1.506
T operator+(const T&, const T&)	3.180	3.063
T operator-(const T&, const T&)	3.140	3.014
T operator*(const T&, const T&)	1.938	2.121
T operator/(const T&, const T&)	8.698	19.616
T operator%(const T&, const T&)	11.038	20.656

检验结果准确性与运行时间代码，在 test.cpp 文件内，可将三个文件添加到工程中运行检测。默认将 BigInteger 类的计算时间与 int 类型数据计算时间比较，若要与 long long 类型计算时间进行比较，可将程序第 11 行

```
1 typedef int Type
```

改为

```
1 typedef long long Type
```

此为第二版本，修复初版以下问题：
1. 缺少 `const BigInteger& operator=(int n)` 赋值函数导致的程序二义性问题
2. 对于常量 `LONG_LONG_MIN abs` 为负值的修正
3. 添加输入与构造函数的 `const char*` 格式检查，格式不符时，当前输入失效，不改变变量值，构造函数则默认构造 `BigInteger(0)`

该类的局限性：
1. 可以与 `bool, int, long long` 数据类型做隐式类型转换（只能从低精度往高精度），但不能与浮点数做隐式类型转换
2. 构造函数不支持 `long int` 类型
3. 对除以 `0` 错误不做处理